Alex Johnson

I pledge my honor that I have abided by the Stevens Honor System.

4a. The sum of i^2 up to n

4b. S = S + i*i

4c. n-1 times

4d. $\theta(n)$

4e. You can store the values computer for each individual iteration in some global array or cache therefore if you were to compute a smaller number n later you could pull it from the stored memory, making the runtime faster.

1a. $x(n) = x(n-1) + 5 \; for \; n > 1, \; x(1) = 0$

$x(n-1) = x(n-2) + 5$

$x(n) = x(n-2) + 10$

$x(n-2) = x(n-3) + 5$

$x(n) = x(n-3) + 15$

$x(n) = x(n-k) + 5k$

$n - k = 1$

$k = n - 1$

$x(n) = x(1) + 5n - 5$

$x(n) = 5n - 5$

1b. $x(n) = 3x(n-1) \; for \; n >, \; x(1) = 4$

$x(n-1) = 3x(n-2)$

$x(n) = 3(3x(n-2)) = 9x(n-2)$

$x(n-2) = 3x(n-3)$

$x(n) = 27x(n-3)$

$x(n) = 3^k x(n-k)$

$n - k = 1$

$k = n - 1$

$x(n) = 3^{n-1} x(1)$

$x(n) = 4 * 3^{n-1}$

1c. $x(n) = x(n-1) + n \; for \; n > 0, \; x(0) = 0$

$x(n-1) = x(n-2) + (n-1)$

$x(n) = x(n-2) + (n-1) + n$

$n(n-2) = x(n-3) + (n-2)$

$x(n) = x(n-3) + (n-2) + (n-1) + n$

$x(n) = x(n-k) + (n-k+1) + (n-k+2) + \cdots + n$

$n - k = 0$

$n = k$

$x(n) = x(0) + \; 1 + 2 + \cdots + \; n$

$x(n) = \frac{n(n+1)}{2}$

1d. $x(n) = x\left(\frac{n}{2}\right) + n \; for \; n > 1, \; x(1) = 1 \; (solve \; for \; n = 2^k)$

$x\left(\frac{n}{2}\right) = x\left(\frac{n}{4}\right) + \frac{n}{2}$

$x(n) = x\left(\frac{n}{4}\right) + \frac{n}{2} + n$

$x\left(\frac{n}{4}\right) = x\left(\frac{n}{8}\right) + \frac{n}{4}$

$x(n) = x\left(\frac{n}{8}\right) + \frac{n}{4} + \frac{n}{2} + n$

$x(n) = x\left(\frac{n}{2^k}\right) + \frac{n}{2^{k-1}} + \frac{n}{2^{k-2}} + \cdots + \frac{n}{2^{k-n}}$

$n = 2^k \quad => \quad k = log\,_2 n$

$x(n) = x(1) + n \sum \frac{1}{2^k}$

$x(n) = 1 + n * \frac{1-\left(\frac{1}{2}\right)^k}{\left(1-\frac{1}{2}\right)}$

$x(n) = 1 + 2n * \left(1 - \frac{1}{2^k}\right) = 1 + 2n - \frac{2n}{2}$

$x(n) = 2n - 1$

1e. $x(n) = x\left(\frac{n}{3}\right) + 1 \; for \; n > 1, x(1) = 1 \; (solve \; for \; n = 3^k)$

$x\left(\frac{n}{3}\right) = x\left(\frac{n}{9}\right) + 1$

$x(n) = x\left(\frac{n}{9}\right) + 2$

$x\left(\frac{n}{9}\right) = x\left(\frac{n}{27}\right) + 1$

$x(n) = x\left(\frac{n}{27}\right) + 3$

$x(n) = x\left(\frac{n}{3^k}\right) + k$

$n = 3^k => k = log\,_3 n$

$x(n) = 1 + log\,_3 n$

3a. $x(n) = x(n-1) + 2 \; for \; n > 0, \;\; x(1) = 0$

$x(n-1) = x(n-2) + 2$

$x(n) = x(n-2) + 4$

$x(n-2) = x(n-3) + 2$

$x(n) = x(n-3) + 6$

$x(n) = x(n-k) + 2k$

$n - k = 1 \;\; => \;\; k = n - 1$

$x(n) = x(1) + 2n - 2$

$x(n) = 2n + 2$

3b. The recursive version of an algorithm and the non-recursive algorithm will be the same because you still have multiply the same number of times in both algorithms.